Analysis and Prediction of Airbnb Listings' Yields in NYC
Project Report

A. **Project Overview**.
Our project seeks to solve the problem of predicting the potential yields (annual earnings) of Airbnb listings in New York City by analyzing publicly available data and building a predictive model.

For this project, we use Inside Airbnb's data on AirBnB listings in NYC (http://insideairbnb.com/get-the-data.html), where publicly available information about a city's Airbnb listings have been scraped and released for independent, non-commercial use. This includes details about the listing such as room type, description, location, price, no. of reviews, etc. Specifically, we will use the latest listings.csv file (last scraped by the provider on October 5, 2020) of the data for listings in NYC.

Our project is structured as: Dataset cleaning and preprocessing -> Dataset Analysis -> Machine Learning

B. **Dataset cleaning/preprocessing**.
Here, we fill/remove missing values, perform datatype conversions, derive new features, remove outliers, perform one-hot encoding, etc.

Datatype conversions were performed for several features to make them usable for visualization and machine learning: price (string to float), amenities (string to int), bathroom_text (string to float), host_since (string to date-time), host_response_time (string to int), host_response_rate (string to float), host_verifications (string to int).

Missing values were either filled or rows with missing values were removed. For example, we fill missing values for 'beds' and 'bathrooms_text' with 1 since listings with no beds or bathrooms are extremely rare or may even be non-existent. We also fill missing values for features 'bedrooms', 'instant_bookable', 'host_has_profile_pic' and 'host_identity_verified'.

We drop rows with missing values if filling said values doesn't make sense. For example, we drop the rows with missing values for the feature 'host_since' since we can't make any reasonable assumptions regarding the time for which such hosts have been active on the platform. We also drop rows with missing values for features 'reviews_per_month', 'review_scores_rating' and 'review_scores_accuracy'.

We derive 3 features from existing features: 'bath_type' denotes the type of the bathroom (private/shared) and was derived from 'bathrooms_text'. 'long term stays' denotes if a listing is available for long term stays (>= 28 days acc. to AirBnB) and was derived from the feature 'maximum_nights'.

Our output feature (the feature we need to predict) is 'yield' which is also a derived feature.
For deriving the yield, we use Inside AirBnB's Occupancy model or San Francisco Model (http://insideairbnb.com/about.html#:~:text=Inside%20Airbnb%20uses%20an%20occupancy,impact%20of%20Airbnb%20on%20housing).
The formula as per said model is: **Average length of stay * price * no. of reviews/month * review rate * 12 months**.
Acc. to Inside AirBnB: the average length of stay in NYC is 6.4 nights (https://blog.atairbnb.com/economic-impact-airbnb/#new-york-city) and the review rate is 0.5 which is constant for all cities.

Features 'price', 'reviews_per_month' and 'yield' have a lot of outliers (Exhibits 1-a, 1-b) which were removed to prevent them from affecting our analysis and prediction. Rows with price less than $400 as well as reviews_per_month less than 6 were removed (Exhibit 1-c) and we are then left with around **96%** of the data (Exhibit 1-d).
We later perform one-hot encoding for converting the following categorical features to numeric: 'host_is_superhost', 'host_has_profile_pic', 'instant_bookable', 'room_type', 'host_identity_verified', 'neighbourhood_group_cleansed'.

C. **Data Analysis**.
Here, we analyze and visualize various features of the dataset and discuss their impact on revenue generation.

First, we analyze the dataset keeping the location, particularly the boroughs, in mind. We see that Manhattan and Brooklyn have the most listings while Staten Island has the least (Exhibits 2-a, 2-b). At this stage we can say that since there will be less competition in Staten Island, hosts may probably earn more money if they have properties there.

Then, we use both price and borough for deeper analysis. We plot the average price for listings as well as the price distribution for each borough (Exhibit 2-c). We observe that Manhattan has the highest average listing price as well as the widest spread of prices, implying that the hosts in Manhattan would face greater competition regardless of the price they charge since there are listings falling under nearly all price ranges. Bronx has the lowest average listing price and the narrowest spread of prices, implying lesser competition since there are fewer listings in Bronx and it will be easier to find an ideal price-point.

Now **we graph the average yield for listings as well as the yield distribution for each borough** (Exhibit 2-d). Staten Island has the highest average listing yield despite having the 3rd highest average listing price, implying a higher booking activity compared to other boroughs, in turn leading to higher yield. Brooklyn has the least average listing yield despite having the most listings in NYC, implying a far less booking activity negatively impacting the yield.

We now use reviews to support our statement about higher and lower booking activity. Guests can post reviews for a listing if and only if they have booked said listing. Thus, **the number of reviews for a given listing is equal to the minimum number of bookings that said listing has had in its lifetime**. For example if a property has 10 reviews, then that property has been booked at least 10 times. Therefore, by plotting the average number of reviews for listings as well as the review distribution for each borough (Exhibit 2-e), we get the average minimum number of bookings the listings in a borough have had.
We see that **average number of reviews is highest for Staten Island which confirms our statement above that on average, Staten Island sees the most bookings. Brooklyn has the 4th lowest average number of reviews which again confirms our statement above that on average, Brooklyn has far fewer bookings in comparison**.
As yield is obviously dependent on the number of times a property is booked, **listings in Staten Island generate high revenues while those in Brooklyn struggle**.

We now move on to **property type's impact on yield**. First we graph the number of each property type per borough (Exhibit 2-f). **Manhattan has all 4 property types**. The **hosts in Manhattan will face greater competition regardless of their property's type**. This combined with the fact that Manhattan has the widest price spread means that the potential for hosts to earn significant revenue in Manhattan is relatively lower due to sheer saturation and increased competition.

Brooklyn also has all 4 property types and the price distribution for its listings is similar to Manhattan's, thus hosts there may also struggle to generate significant yield.

Queens, Bronx and Staten Island seem to be much more lucrative. **Bronx has no hotel rooms** and **Staten Island only has entire homes and private rooms.** For people who own a hotel in either of these boroughs there will be no existing competition. **Queens also has all 4 property types but its lower average listing price makes it attractive to potential guests** and could thus lead to more bookings which in turn will lead to substantial revenue. In this case, there are hotel rooms in Queens but very few and thus low competition.

We now use price and property type for deeper insights. We plot the average price and the price distribution for each property type (Exhibit 2-g). We observe that **a hotel room 's average price is significantly higher** than all other property types while that of the shared room is the least. Further, we plot the average yield, yield distribution and average number of reviews (to get the minimum average number of bookings) for each property type (Exhibits 2-h, 2i).
Hotel rooms have the highest average yield and **shared rooms the lowest**. **Hotel rooms have the least average number of reviews and thus the least average minimum bookings.** But yet their **yield on average is the highest due to their high prices.** Since not everyone owns a hotel, a private room or an entire home/apt would make more sense for most hosts. A private room and an entire home are booked more often**, an entire home/apt has the 2nd highest average yield** as it has the 2nd highest average price and the highest minimum number of bookings**.** A **private room has the 3rd highest average yield** as it has the 3rd highest average price and the 2nd highest minimum number of bookings**.** A **shared room has the lowest yield** as it is booked less often. Thus, **a hotel room is most lucrative but for those who do not own a hotel room, a private room or an entire home will be ideal**.

Moving on to **analyze the impact of the host type (superhost or not) on yield**. We again plot the average price, yield and number of reviews as well as their distributions for both ordinary and super hosts (Exhibits 2-j, 2-k, 2-l, 2-m).
We see that **super hosts charge a higher price on average but the difference is not drastic** and yet, **t**he **average yield of listings with a superhost is more than twice than those with ordinary hosts**. This means that people book super hosts' properties more and this is confirmed when we plot the average number of reviews for both host types. **Listings with super hosts receive twice as many reviews** on average and therefore, are **booked at least twice as many times as ordinary hosts**. Hence, a host can **increase revenue by becoming a superhost.**

We analyze the annual availability (number of days/year that a listing is available for booking) (Exhibits 2-n, 2-o, 2-p). The dataset shows us that **more than 40% of the listings are not available for booking**, understandably so due to the pandemic. However, this can work in a potential hosts' favour as well as they can potentially get many more bookings and thus earn more revenue provided there is a healthy demand from the guests' side. But even in case the demand from the guests' side declines, it is still beneficial to keep one's listing active. The demand despite being reduced, never went to 0 outright and thus, there is a chance that a person and/or people may end up booking the property.

Next we **visualize reviews per month and reviews per month (ltm)** (Exhibit 2-q) to gain insights as to how they affect the revenue. The graphs tell us that the distributions of the said features are highly skewed in the data set and we can see a high concentration in the lower values. Also, both graphs clearly illustrate **a linear relationship between the features and yield**, confirming that higher the number of reviews, higher the yield.

Finally, we investigate whether there is a relationship between accommodates, bathrooms as well as bedrooms with the yield (Exhibits 2-r, 2-s).
Even though the number of people that a property can accommodate will obviously affect its earning potential, the graphs show **no clear relationship between accommodates and yield**.
For example, properties that can accommodate 6 people have both a lower as well as a higher yield than properties which accommodate 8.
This may be because even if a property can accommodate a lot of people, potential guests may not be travelling in large groups and thus would go for listings that accommodate fewer people. The opposite is also possible since a large group of travelers will most likely want to book a listing that can accommodate several.

Again, there is **no clear relationship between both bedrooms or bathrooms and yield**. For example, we see that some properties with 4 bedrooms have lower yields than those with 5 bedrooms while some have higher yields. Similarly, some properties with 3 bathrooms have both lower yields than those with 4 bathrooms while some have higher yields.
Obviously the number of bedrooms and bathrooms affect the price of the listing but just like in the case of accommodations, potential guests may or may not be travelling in large groups and would choose a property depending on that.

Therefore we can say that 'accommodates', 'bathrooms' and 'bedrooms' do not have a high predictive power in this case but they are certainly important factors and thus must be used in conjunction with other features. This is exactly what happens when we use machine learning.

D. **Machine Learning**
We start by first plotting a correlation matrix to visualize the relationship between features in the dataset and then categorize the features into 3 categories (greatly positively correlated, positively correlated, negatively correlated) based on their correlation with yield. (Exhibits 3-a,3-b).

Then we proceed with the training: **we perform both cross-validation (k value = 10)** and **hold-out testing** on 3 models, viz., Linear Regression, Decision Tree, Random Forest.

We remove yield from the training data as it is the output variable. **We also remove price and reviews per month** as they were used to calculate the yield and therefore have a very strong correlation with it.

Metrics used are Mean Squared Error (MSE) Loss and r-squared (r^2) variance score. MSE allows for an intuitive measure of error i.e. an MSE of 10000 indicates the yield predicted by the model is off by USD 100 (square root of 10000). Thus, **lower the MSE value, the better**.
R-squared score measures how well the model fits the data and **for r-squared score, higher the percentage, the better is the fit**.
Thus a well performing model will have a low MSE loss value and a high r^2 score.

Early on during data preprocessing, we had removed outliers from the dataset (rows with price < 400 and reviews per month <6). Sometimes removing outliers and thus reducing the dataset can negatively affect the model's performance.
We use cross-validation to check whether the models perform better with or without outliers and if outliers are removed then exactly how much data should we remove. **Specifically, we see what percent (from 96% to 100%) of the original data gives us the best results**.
Of course in addition to this, cross-validation also serves to evaluate the models.

We perform cross-validation 4 times:

- Remove outliers for the price feature and evaluate r2 score.

- Remove outliers for the price feature and evaluate MSE.
- Remove outliers for the reviews per month feature and evaluate r2 score.
- Remove outliers for the reviews per month feature and evaluate MSE score.

After cross-validation, we clearly see a performance improvement due to removal of outliers and Random forest performs best (Exhibits 3.c, 3.d). We also see that acc. to cross-validation, **our initial removal of outliers by removing rows with prices < 400 and reviews per month < 6 that left us with 96% of the data, was correct**.

Now, we perform hold-out testing. Each of the 3 models are **trained for 10 epochs** with a **70:30 train-test split using the ideal percentage of the dataset that we found using cross-validation**. After training, we plot the loss and variance scores for all 3 models (Exhibits 3-e, 3-f, 3-g, 3-h). We note that **random forest performs best** while **linear regression performs the worst, same as cross-validation**.

Now we plot and analyze feature importance for all 3 models (Exhibits 3-i, 3-j, 3-k), starting with random forest.
Most of the features that Random Forest considered to be important share a positive correlation with yield (refer Exhibits 3-b, 3-i). The most important feature is **no of reviews ltm** (ltm: last twelve months) which, according to the correlation matrix, **is greatly positively correlated** with yield and same is the case for **No. of reviews**. This makes intuitive sense as the number of reviews that a property has received recently (in the past year) and its total lifetime reviews reflect the property's as well as the host's quality which of course is a factor that potential guests take into consideration.
We can also make out that the algorithm considers **accommodates** and **room type** to be highly important (they **are both positively correlated with yield**). This also makes sense as the number of people a property can accommodate as well as the property's type greatly influence a potential guest's decision to book a listing and therefore has great impact on the earning potential of that property.
No. of amenities is also considered important as well as it **is positively correlated.** Rightfully so since properties with more amenities would attract more guests as they provide more "bang for the buck".
Random Forest successfully recognized the features with the most predictive power and was thus able to achieve a low loss value as well as a high variance score.

Linear regression failed to give enough importance to features greatly positively correlated with yield and instead ended up emphasizing on features with lower correlation values.
The algorithm considers **No of reviews ltm** and **No. of reviews** to be less important while giving more than necessary importance to **instantly bookable** and **profile photo** (refer Exhibits 3-b, 3-j). This means that **Linear Regression was unable to recognize features with the most predictive power** and therefore achieved a high loss value as well as a low variance score.

Decision Tree gives enough importance to some positively correlated features such as accommodates, room type, no. of reviews ltm, etc. However, **Decision Tree also fails to give any importance at all to several other positively correlated features** such as **host response rate**, **superhost**, etc. (refer Exhibits 3-b, 3-k).This means that **Decision Tree recognizes some but not most of the features with high predictive power** and therefore, it performs better than Linear Regression but is not as good as Random Forest.

Finally, we take the best performing model that is Random Forest and briefly discuss its performance. Since the model was trained for 10 epochs, it has 10 MSE loss values. So, **in order to get a conservative estimate, consider the worst-case scenario, that is, the highest loss value**. After that, we **take the square root of that loss value to get the dollar value by which the predicted yield was off**.

Given more time, with additional data, more training as well as hyper-parameter tuning, we can reduce loss and improve the results.

Appendix

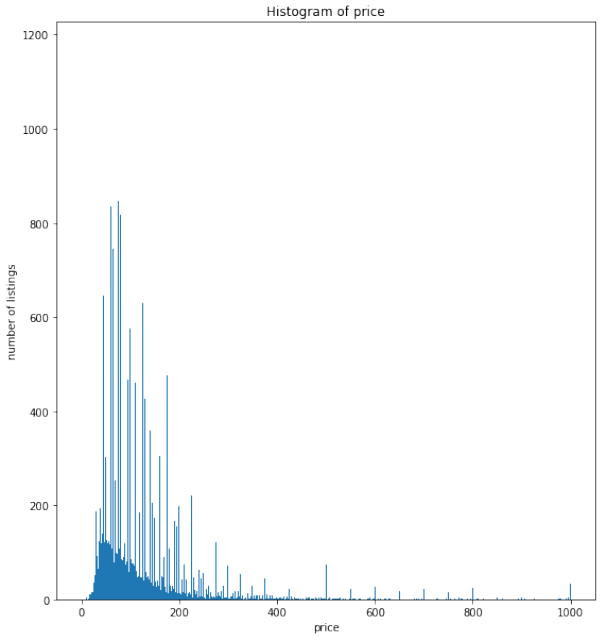
Exhibit 1-a Statistics for features: price, reviews_per_month and yield.

```
df[['price','reviews_per_month','yield']].describe()
```

	price	reviews_per_month	yield
count	34006.000000	34006.000000	3.400600e+04
mean	137.899739	0.910760	4.410969e+03
std	240.061344	1.282411	1.041874e+04
min	9.000000	0.010000	1.152000e+01
25%	65.000000	0.120000	4.792320e+02
50%	100.000000	0.390000	1.585152e+03
75%	155.000000	1.210000	4.961952e+03
max	10000.000000	44.060000	1.104861e+06

These outliers will affect the overall analysis as well as prediction and thus need to be eliminated.

Exhibit 1-b Plot of features price and reviews_per_month.



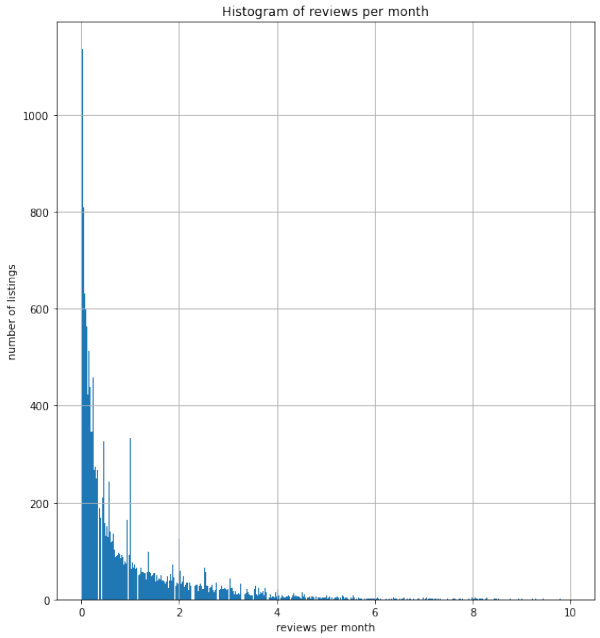


Exhibit 1-c Statistics for features: price, reviews_per_month and yield after removing outliers.

	price	reviews_per_month	yield
count	32771.000000	32771.000000	32771.000000
mean	116.580254	0.861460	3724.341015
std	71.005082	1.077132	5595.879392
min	9.000000	0.010000	11.520000
25%	65.000000	0.120000	460.800000
50%	99.000000	0.390000	1479.936000
75%	150.000000	1.190000	4608.000000
max	399.000000	5.990000	66831.744000

Exhibit 1-d Percentage of data retained after removing outliers.

```
print("Percentage of the dataset after removing outliers: ", len(preprocessed_df2.index)/len(df.index))
```

```
Percentage of the dataset after removing outliers:  0.9636828794918544
```

Exhibit 2-a Airbnb listing distribution graphs.

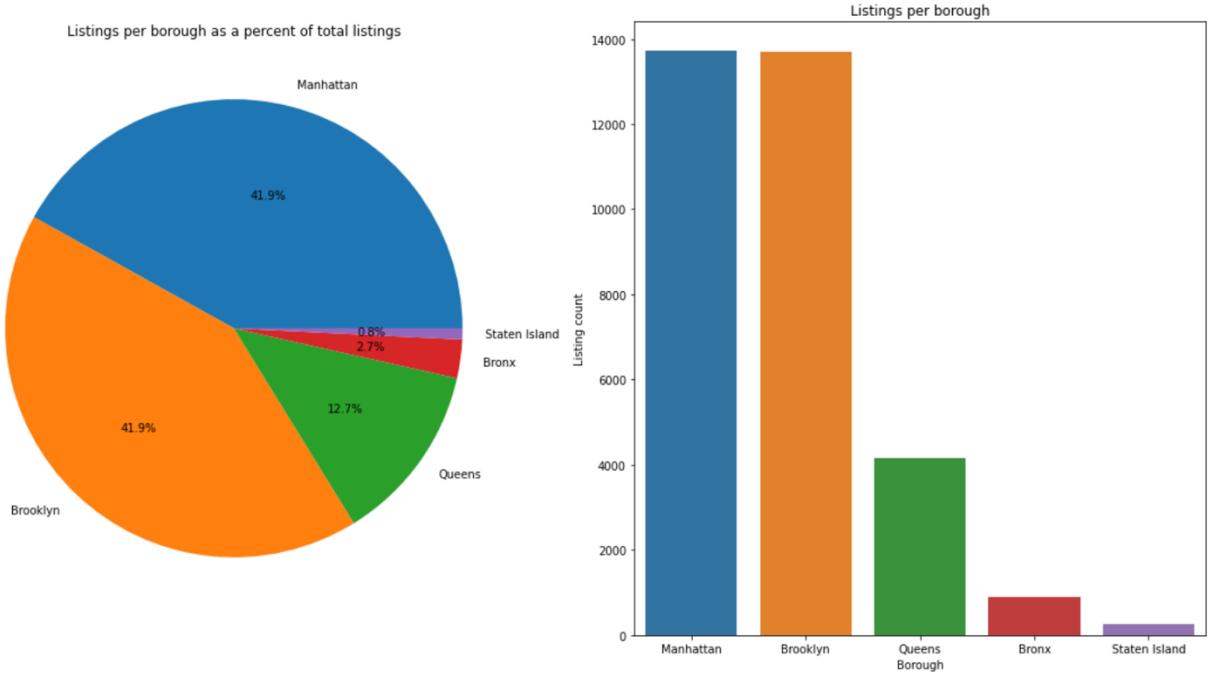


Exhibit 2-b Airbnb listing distribution table.

```
Manhattan        13721
Brooklyn         13701
Queens            4156
Bronx              889
Staten Island      256
Name: neighbourhood_group_cleansed, dtype: int64
```

Exhibit 2-c Average price and pricing distribution per borough.

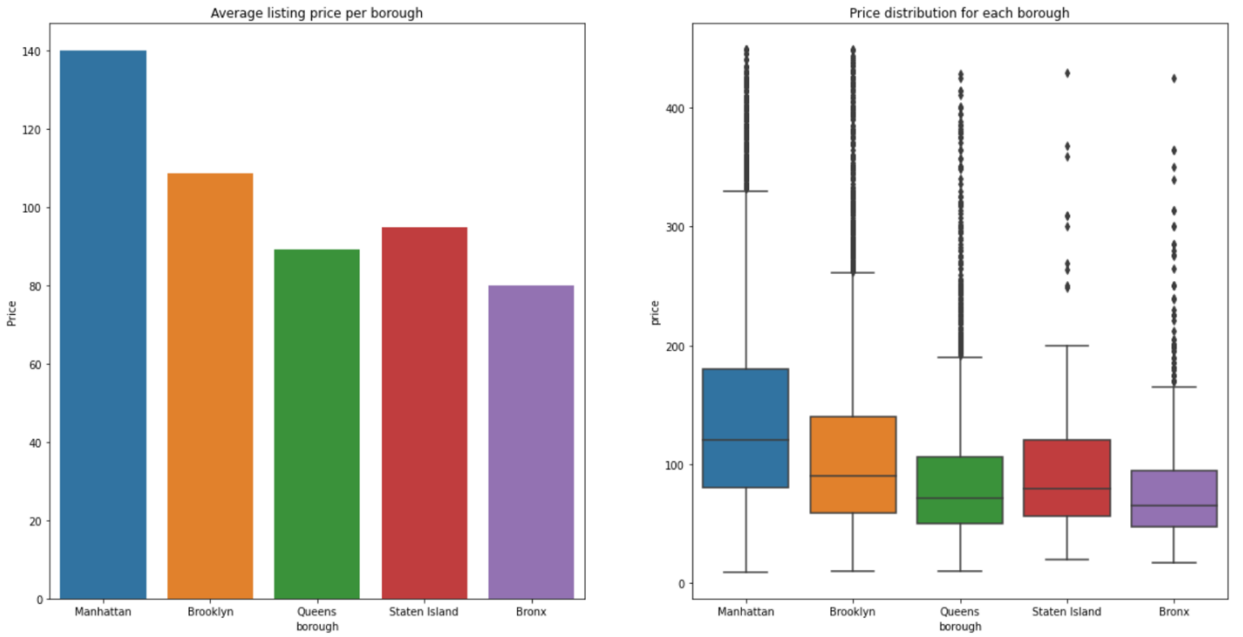


Exhibit 2-d Average yield and yield distribution per borough.

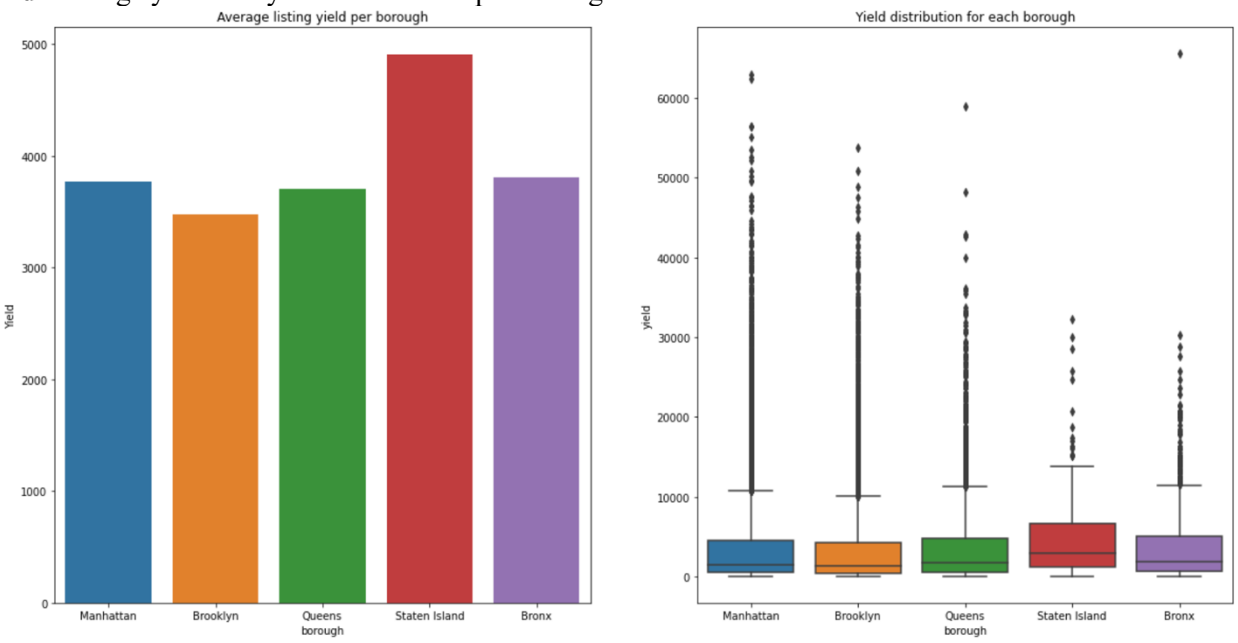


Exhibit 2-e Average review and review distribution per borough

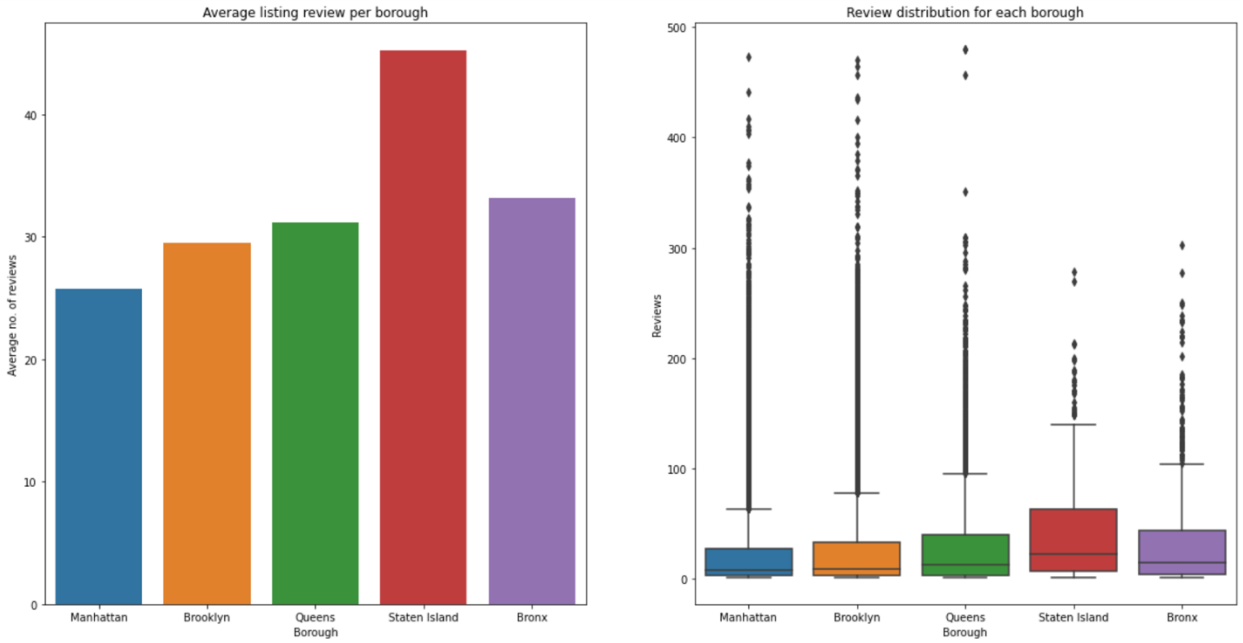


Exhibit 2-f Property type distribution per borough

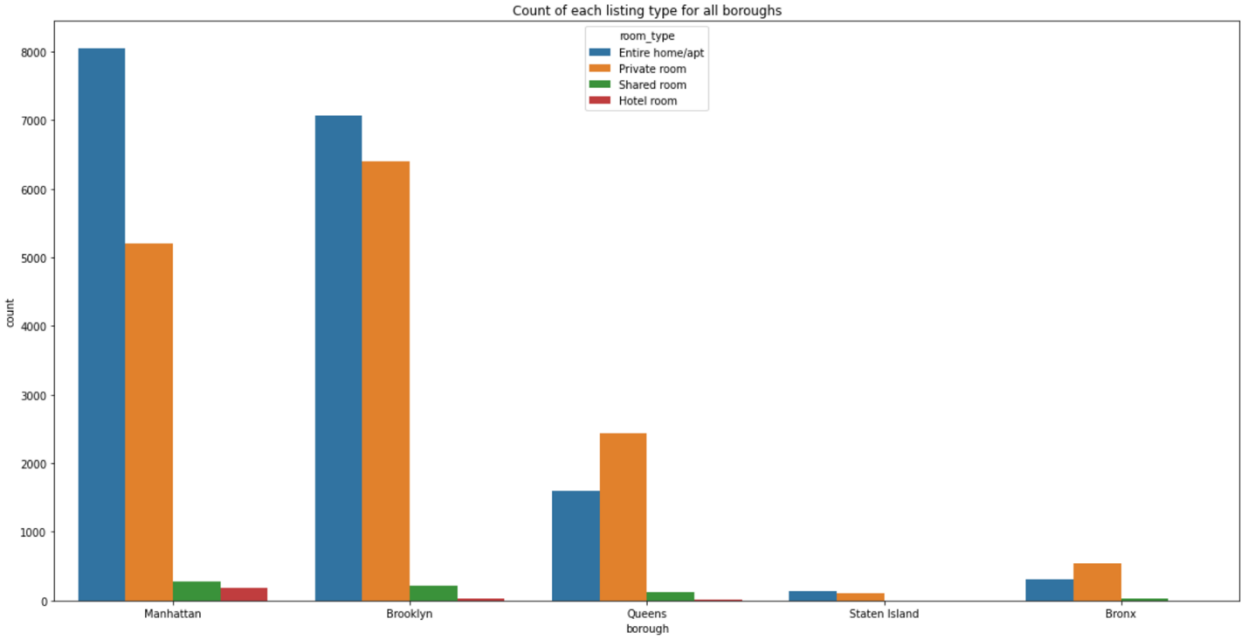


Exhibit 2-g Average price and price distribution per property type

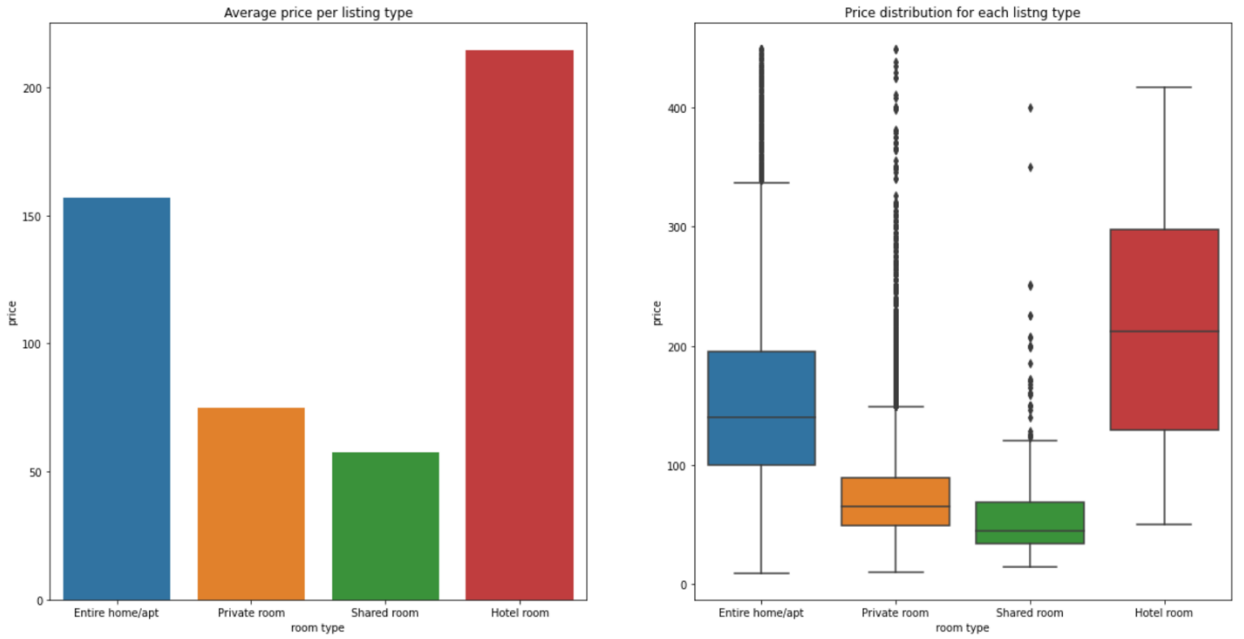


Exhibit 2-h Average yield and yield distribution per property type

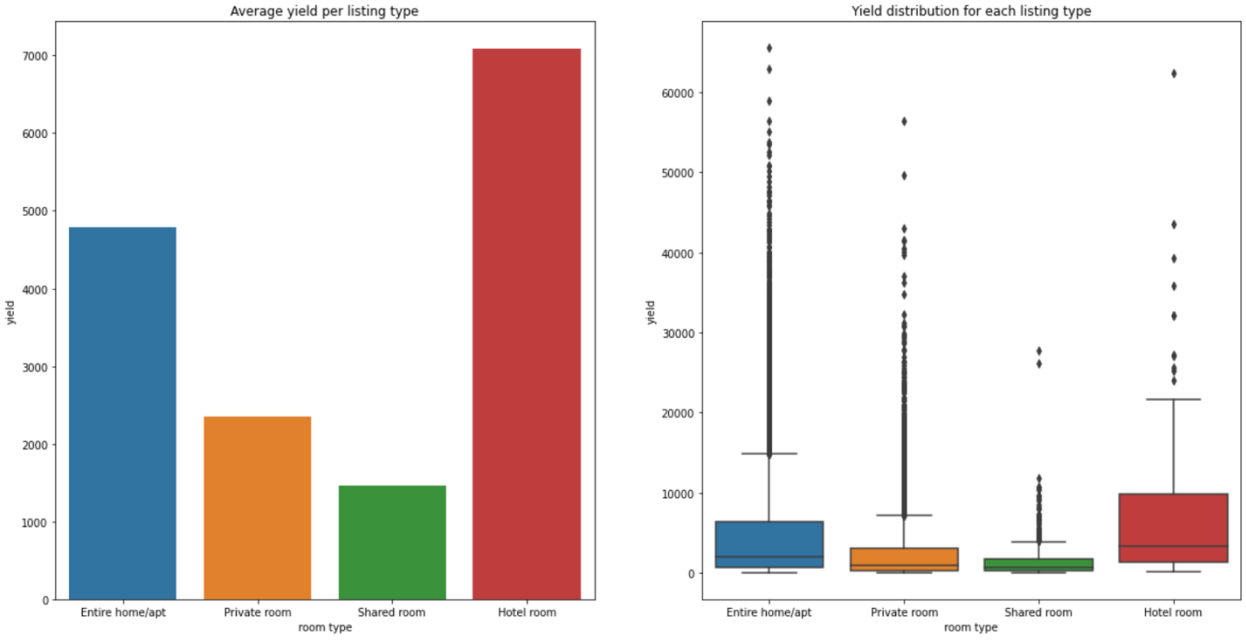


Exhibit 2-i Average review and review distribution per property type

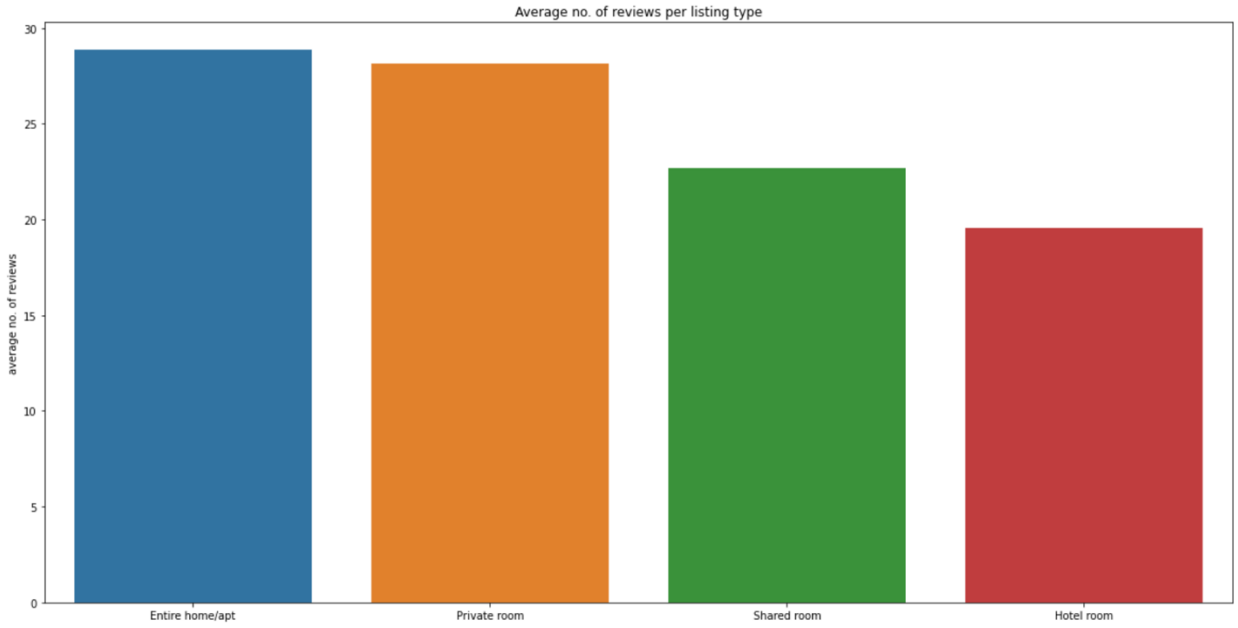


Exhibit 2-j Superhost distribution map

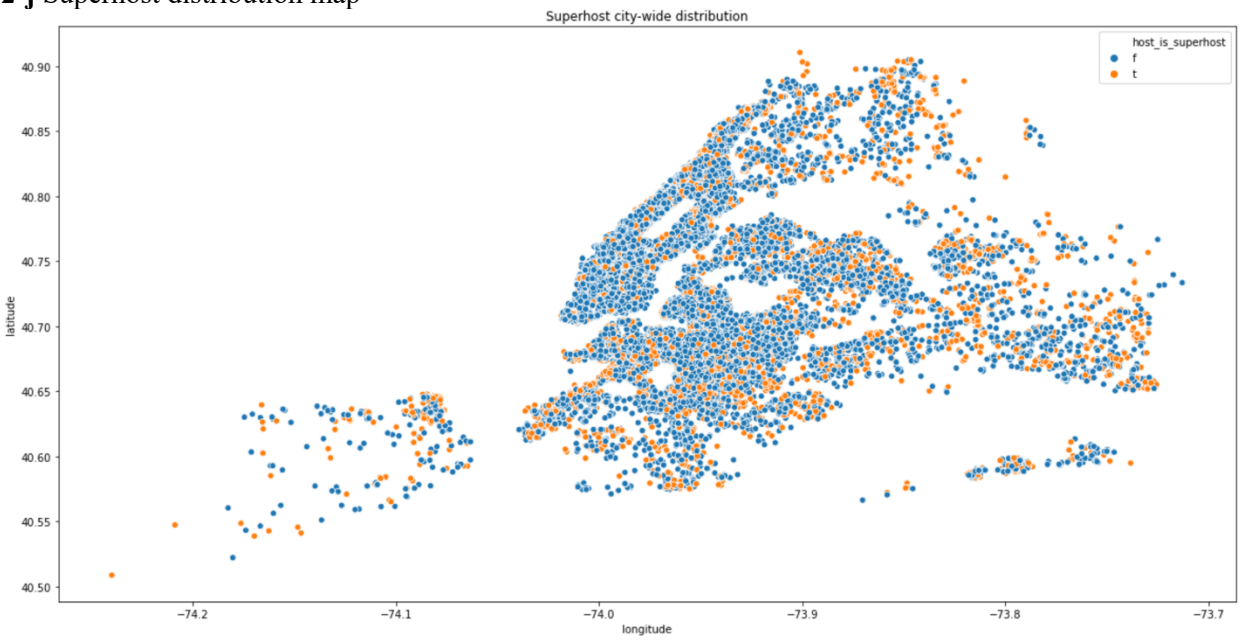


Exhibit 2-k Average price and price distribution per host type

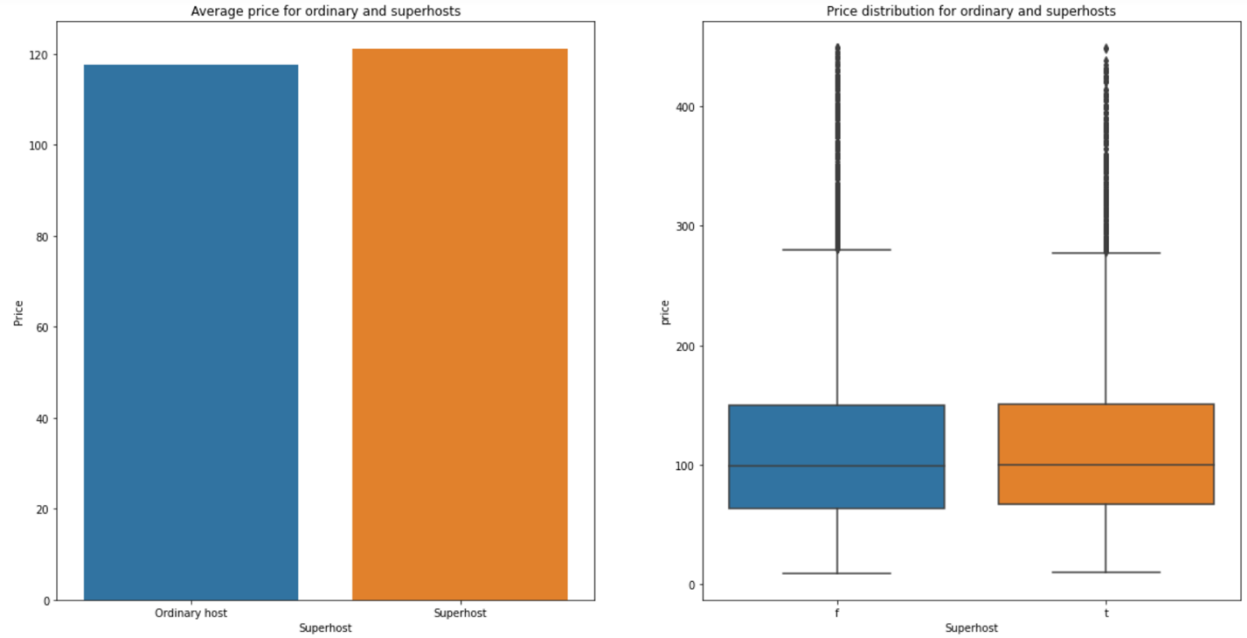


Exhibit 2-l Average yield and yield distribution per host type

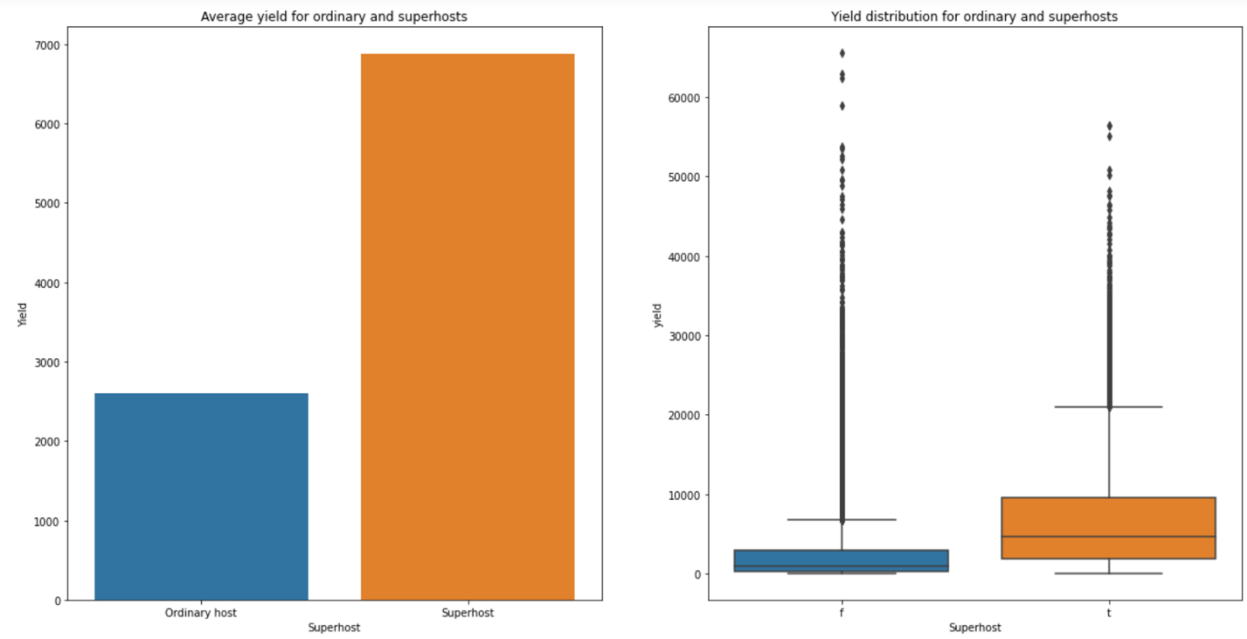


Exhibit 2-m Average review and review distribution per host type

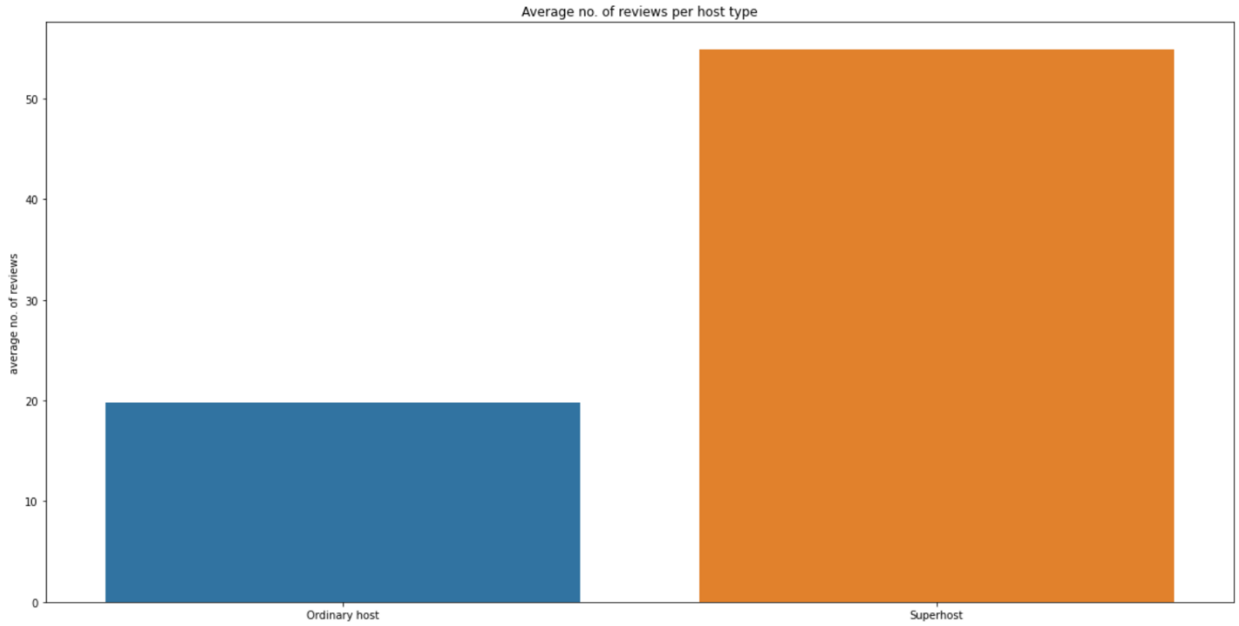


Exhibit 2-n Average review and review distribution per host type

But the dataset shows us something beyond the obvious and much more relevant to the current scenario. **We can see below that we have listings that are available for 0 days. That is, they are unavailable throughtout the year**.

```
preprocessed_df['availability_365'].describe()
```

```
count    32723.000000
mean       120.549094
std        141.527969
min          0.000000
25%          0.000000
50%         56.000000
75%        251.000000
max        365.000000
Name: availability_365, dtype: float64
```

Now let's see how many listings are available for at least 1 day.

```
available_listings = preprocessed_df[preprocessed_df['availability_365']>0]
len(list(available_listings['availability_365']))
```

```
18560
```

Now let's see how many listings are unavailable, that is, 0 days.

```
unavailable_listings = preprocessed_df[preprocessed_df['availability_365']==0]
len(list(unavailable_listings['availability_365']))
```

```
14163
```

Thus, we see that more than 40% of all the listings in NYC are unavailable.

Following are the plots for both available and unavailable listings.

Exhibit 2-o Distribution of available listings

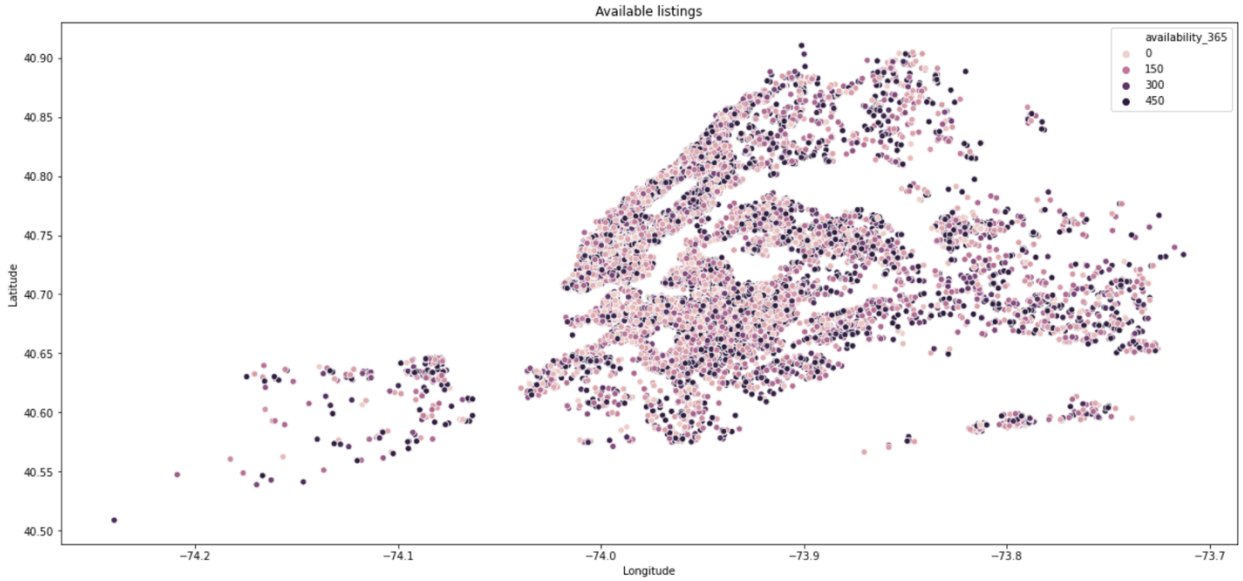


Exhibit 2-p Distribution of unavailable listings

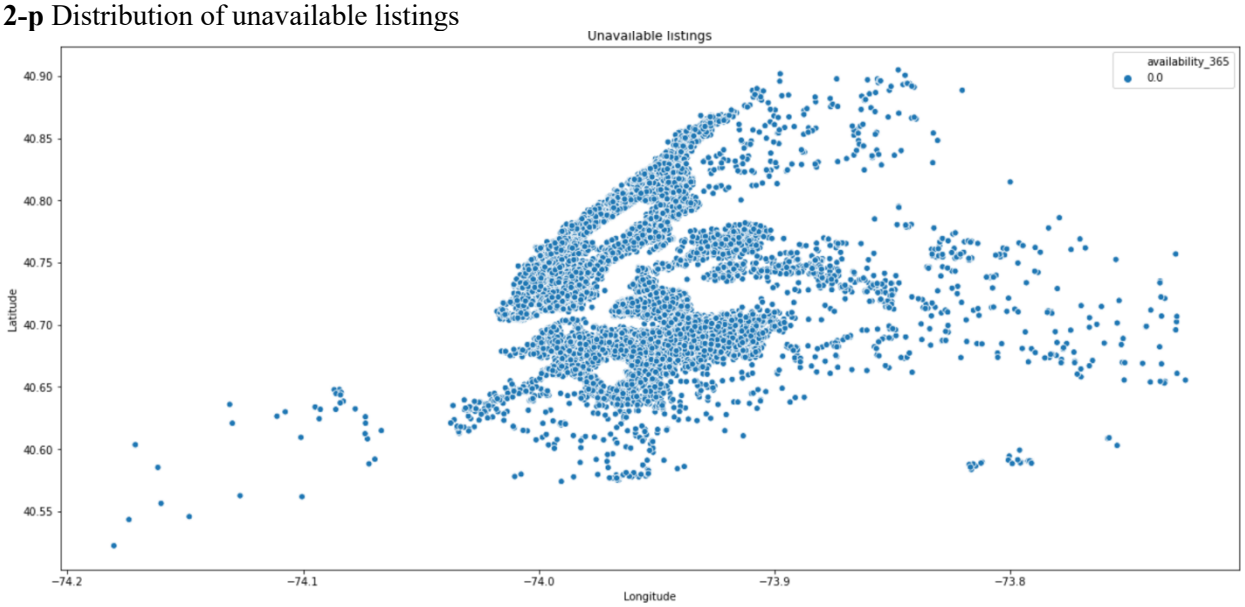


Exhibit 2-q Reviews per month and reviews per month (ltm) vs yield

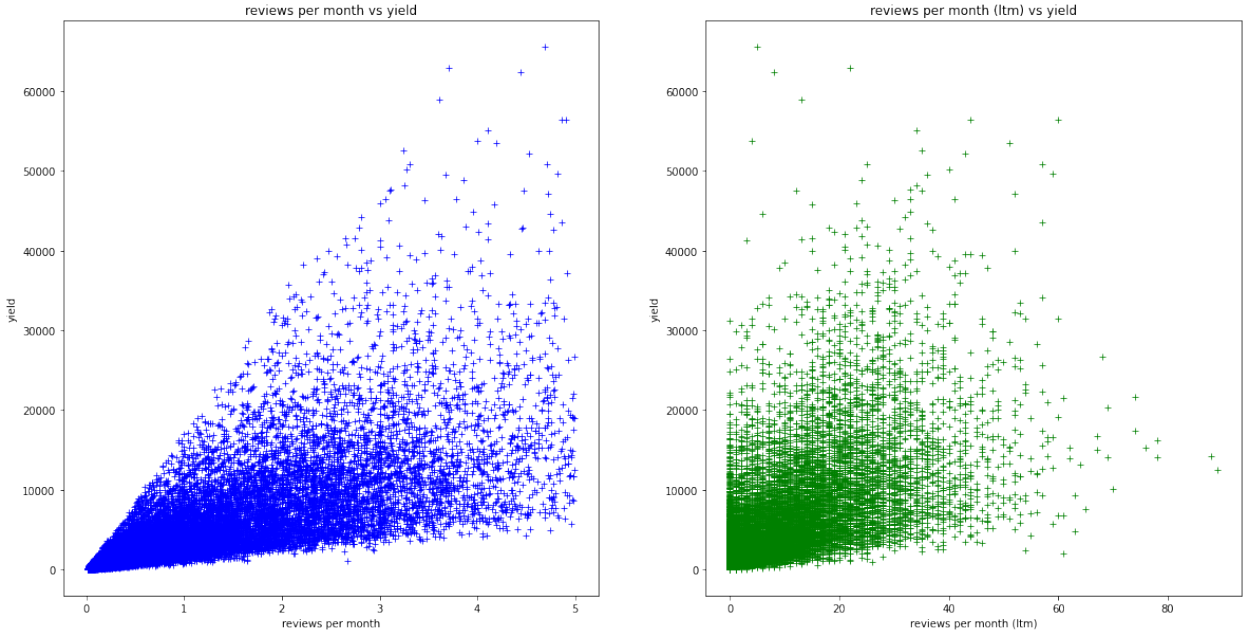


Exhibit 2-r Accommodates vs yield

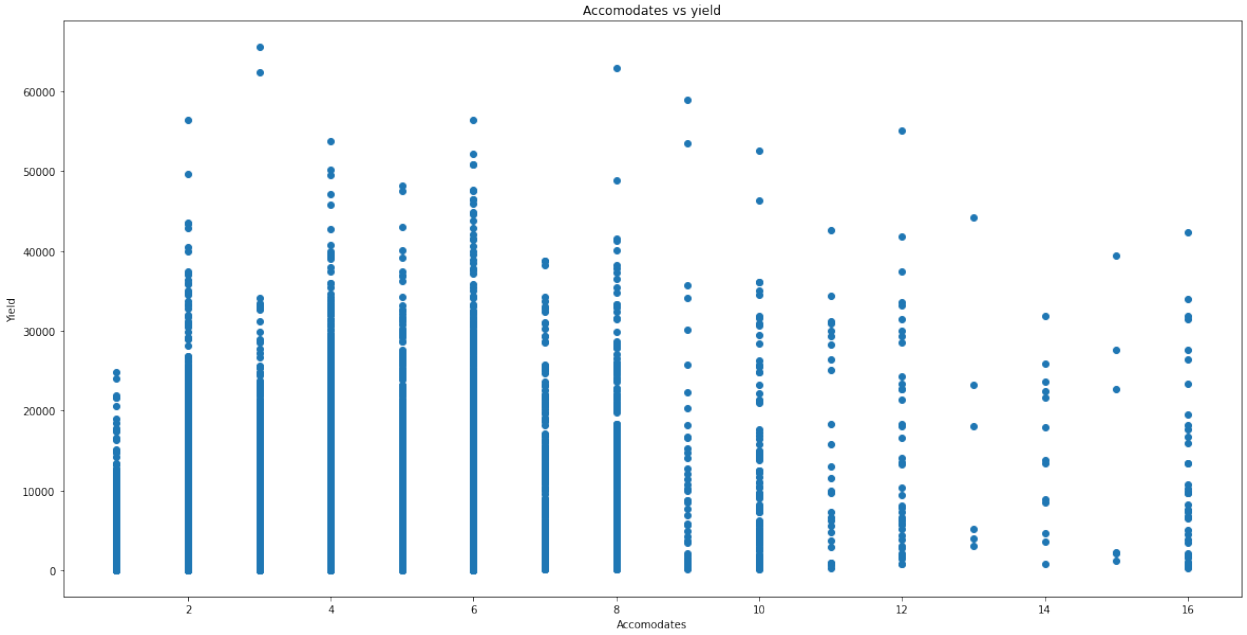


Exhibit 2-s Bathrooms and bedrooms vs yield

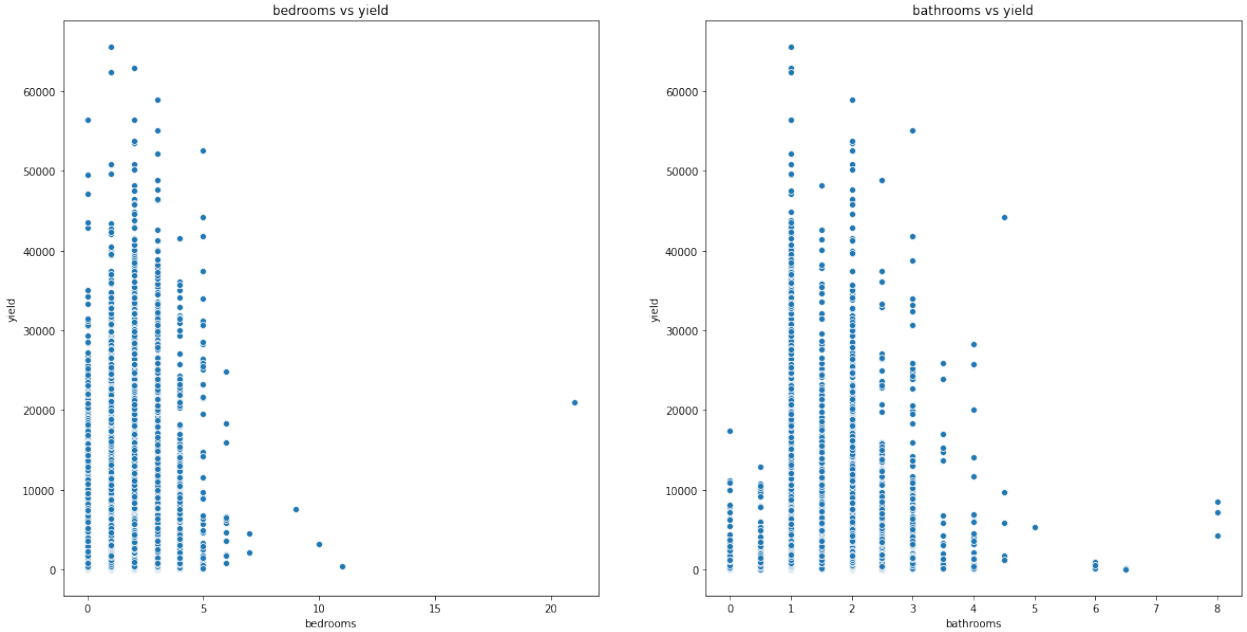


Exhibit 3-a Correlation matrix

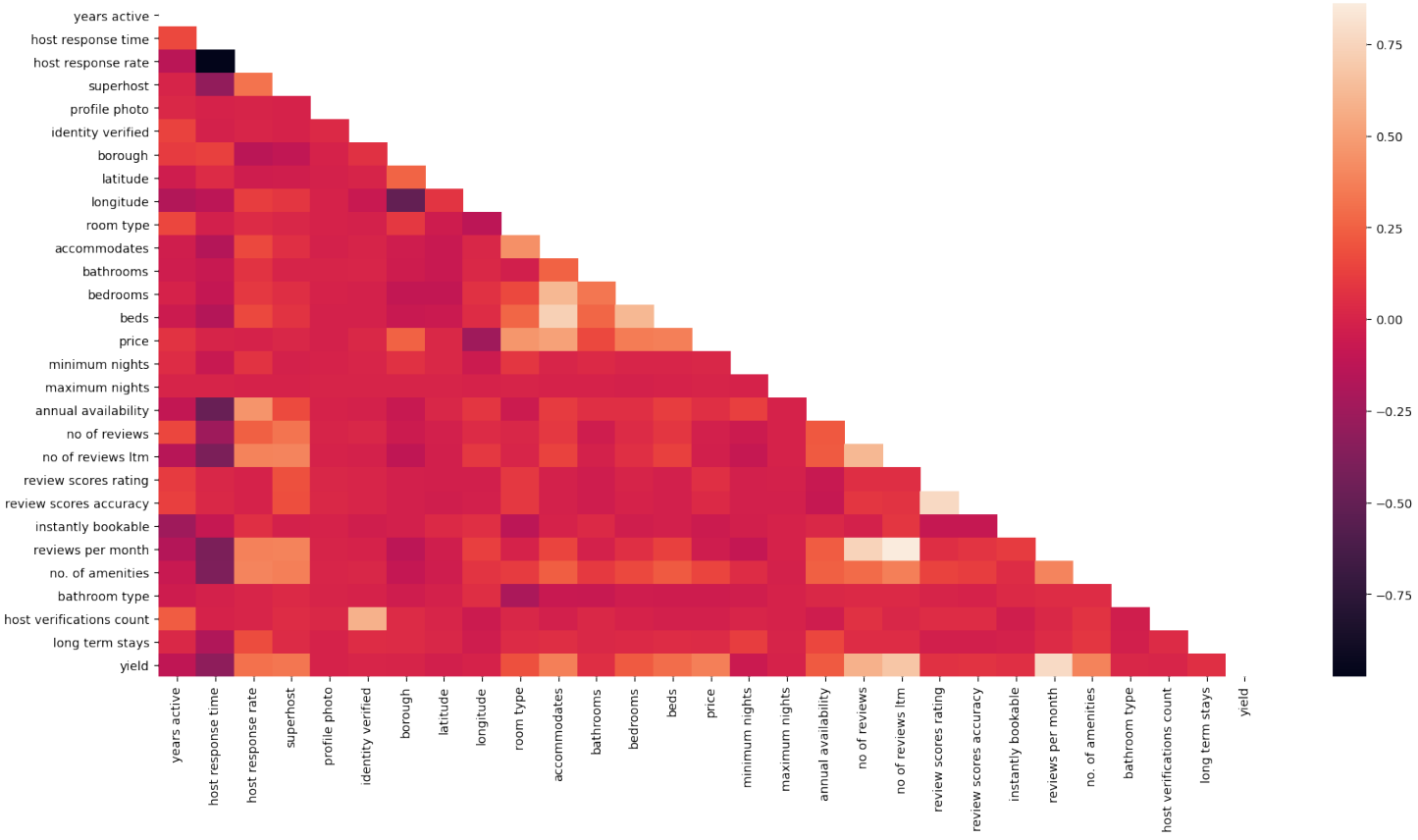


Exhibit 3-b Feature categorization based on correlation

	greatly positively correlated	positively correlated	negatively correlated
0	no of reviews	host response rate	years active
1	no of reviews ltm	superhost	host response time
2	reviews per month	identity verified	profile photo
3		borough	latitude
4		room type	longitude
5		accommodates	minimum nights
6		bathrooms	maximum nights
7		bedrooms	
8		beds	
9		price	
10		annual availability	
11		review scores rating	
12		review scores accuracy	
13		instantly bookable	
14		no. of amenities	
15		bathroom type	
16		host verifications count	
17		long term stays	

Exhibit 3-c Cross-validation evaluation after removing outliers in price feature

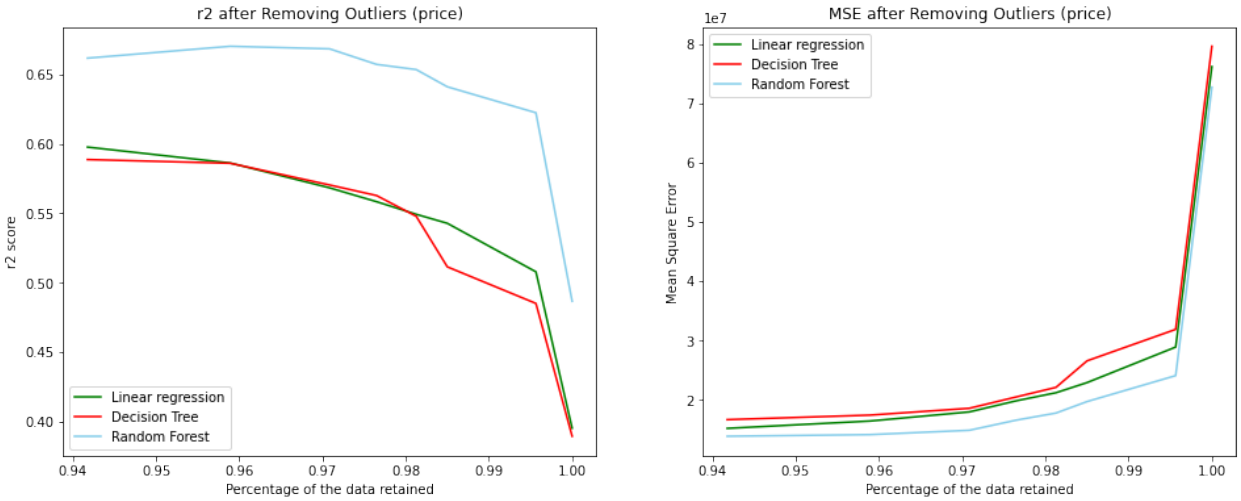


Exhibit 3-d Cross-validation evaluation after removing outliers in reviews per month feature

Exhibit 3-e MSE Loss graph for all 3 models.

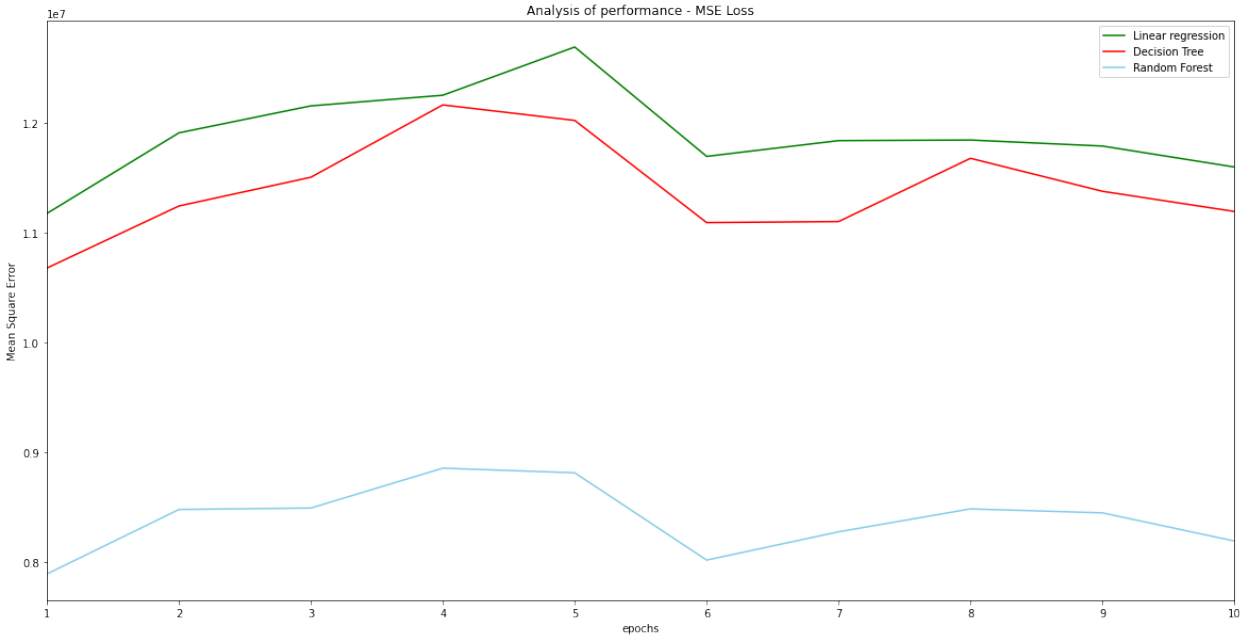


Exhibit 3-f r^2 score graph for all 3 models.

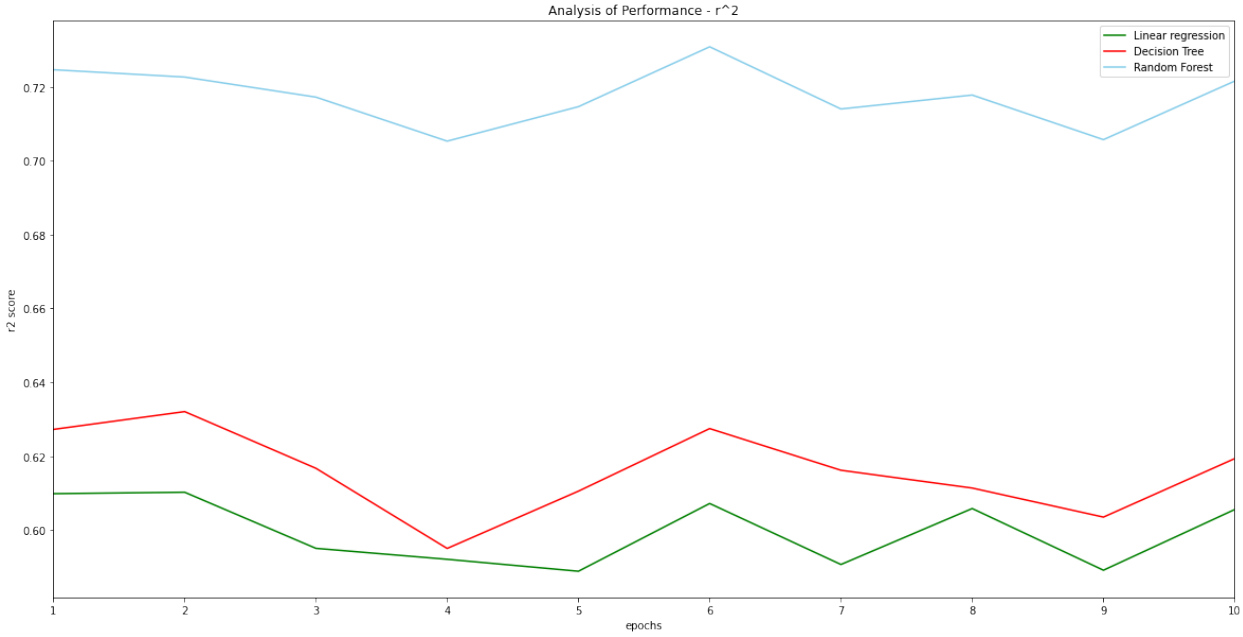
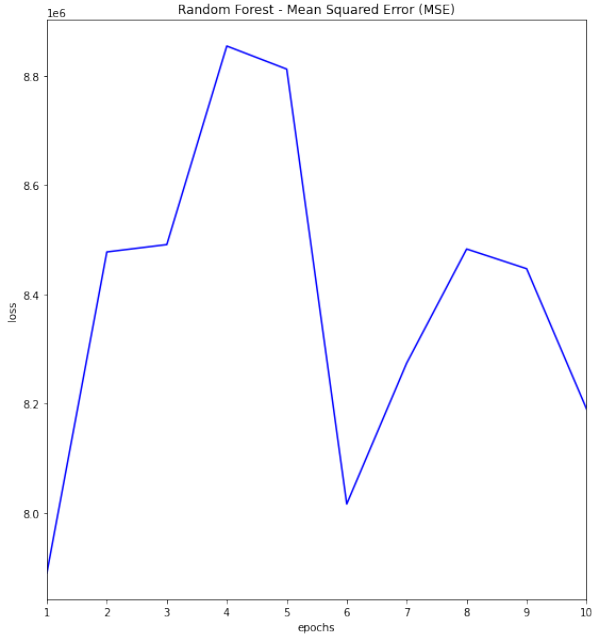


Exhibit 3-g Performance metrics for all 3 models.

	Random Forest MSE	Random Forest r^2	Decision Tree MSE	Decision Tree r^2	Linear Regression MSE	Linear Regression r^2
1	7.890329e+06	0.724594	1.067914e+07	0.627253	1.117731e+07	0.609865
2	8.477666e+06	0.722609	1.124456e+07	0.632076	1.191155e+07	0.610252
3	8.491150e+06	0.717176	1.150664e+07	0.616736	1.215720e+07	0.595068
4	8.854496e+06	0.705271	1.216609e+07	0.595041	1.225425e+07	0.592107
5	8.812200e+06	0.714594	1.202478e+07	0.610546	1.269393e+07	0.588874
6	8.016397e+06	0.730802	1.109318e+07	0.627481	1.169681e+07	0.607211
7	8.274317e+06	0.713973	1.110258e+07	0.616206	1.184047e+07	0.590698
8	8.483052e+06	0.717739	1.167924e+07	0.611391	1.184593e+07	0.605845
9	8.446996e+06	0.705674	1.137863e+07	0.603524	1.179110e+07	0.589152
10	8.190225e+06	0.721516	1.119477e+07	0.619355	1.160004e+07	0.605575

Exhibit 3-h MSE and r^2 score graph for random forest.



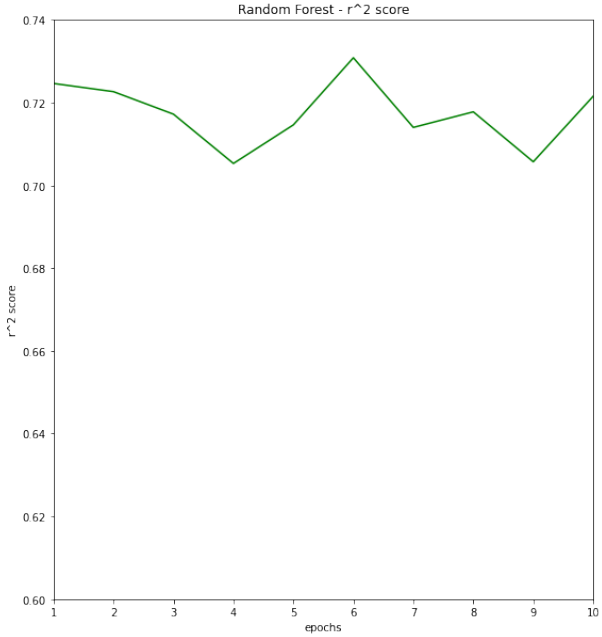


Exhibit 3-i Feature importance for random forest.

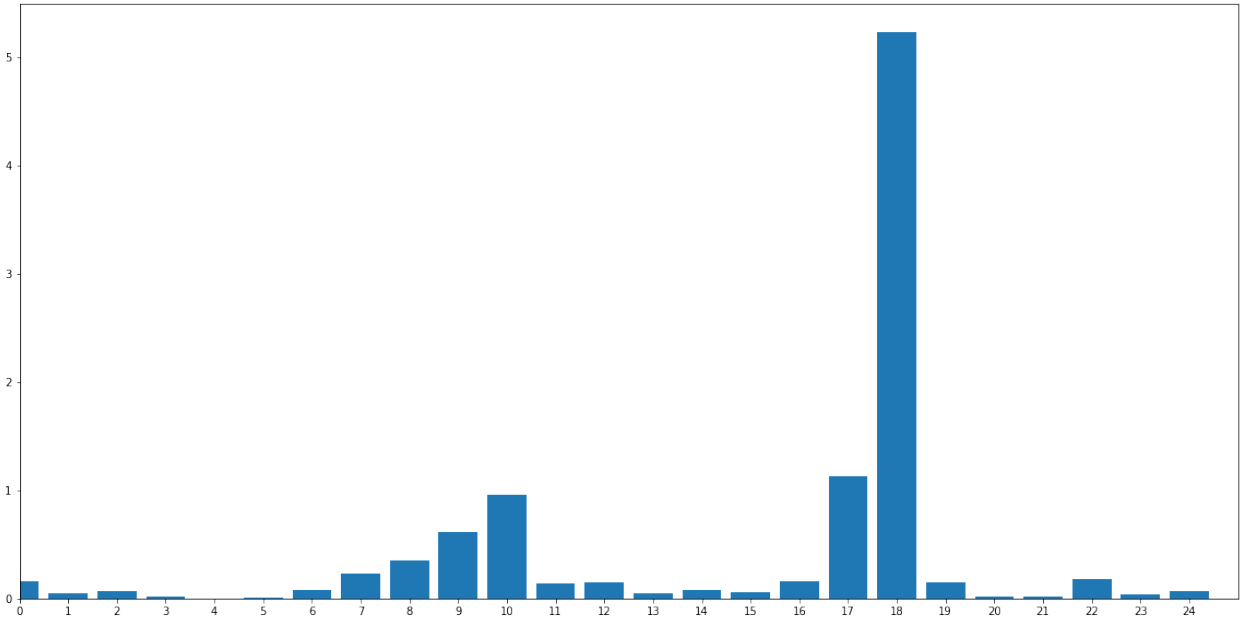


Exhibit 3-j Feature importance for Linear Regression.

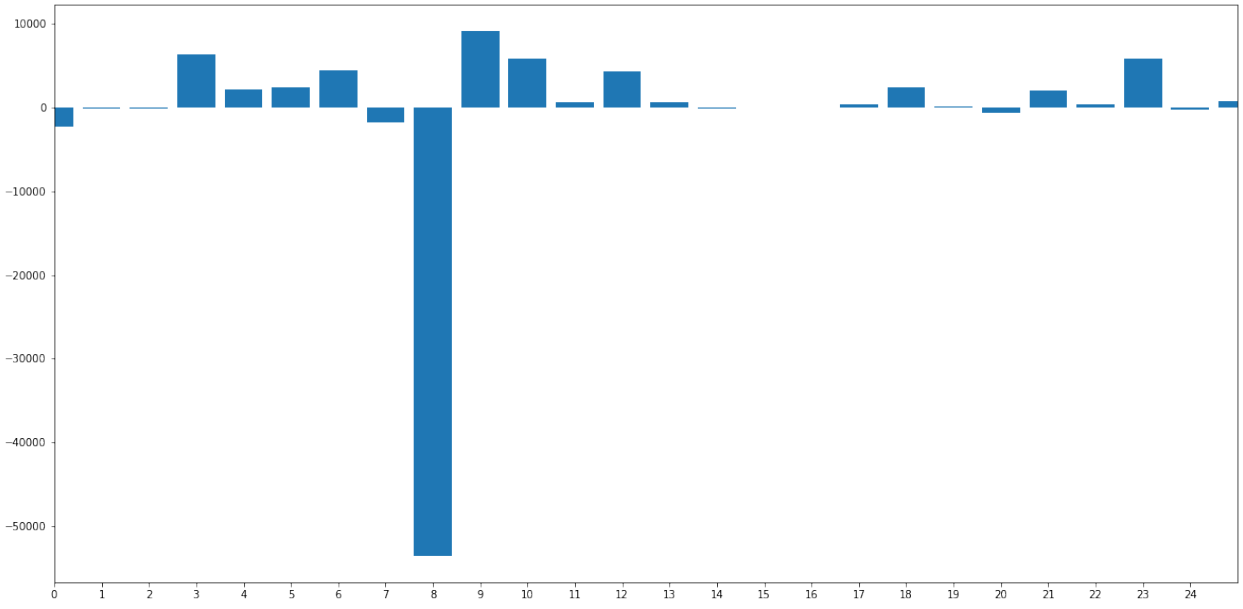


Exhibit 3-k Feature importance for Decision Tree.